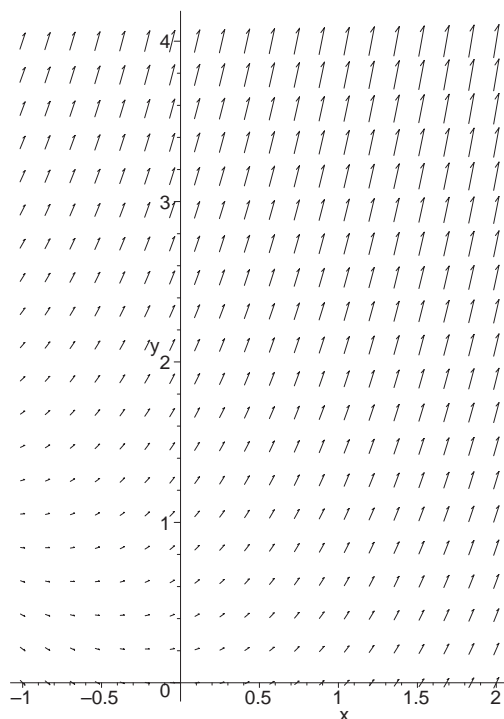


Direction Fields and Euler's Method

Direction Fields As we begin our study of differential equations we come across equations of the form $y' = f(x, y)$, which means that at each point in some region the slopes of solutions of the equation are known. It is very useful to be able to plot short line segments at a reasonable number of points and thereby have a picture of the behavior of these slopes. The command we need in Maple is found in the package 'plots' and the problem is that the input must have two coordinates, not one. We can get around this by making the first coordinate equal to 1 and the second the value of $f(x, y)$.

Maple Example: Suppose we are given a direction field $y' = x + y$ and we wish to plot it.

```
> with(plots):
> f:=(x,y)->x+y;
                                     f := (x, y) → x + y
> F:=(x,y)->[1,f(x,y)];
                                     F := (x, y) → [1, f(x, y)]
> fieldplot(F(x,y),x=-1..2,y=0..4);
```



Euler's Method This method assumes that a direction field, $y' = f(x, y)$, and a boundary value, $y_0 = y(x_0)$, are known for a potential solution y . Starting at x_0 , a linear approximation is used to locate a second point and this process is repeated until an approximate solution for the equation is found for $x = x_n$. Let's begin by recalling the *point-slope* form for the equation of a line:

$$y - y_0 = m(x - x_0)$$

Identify $h = x - x_0$ and $m = f(x_0, y_0)$ and you have

$$y = y_0 + f(x_0, y_0)h \quad \text{or} \\ y_{i+1} = y_i + f(x_i, y_i)h$$

when this is viewed as an iterative process. As you will soon see, spreadsheets are quite useful here.

Maple Example Let's consider the direction field from our example above and the boundary value $y(0) = 1$ and approximate the value of $y(1)$ by using a spreadsheet. For simplicity, we will redefine f in this worksheet. The readers are urged to do the steps on their computers as they read along.

```
with(plots):
> f:=(x,y)->x+y;
```

$$f := (x, y) \rightarrow x + y$$

Click on **Insert** and **Spreadsheet** and resize the spreadsheet so that there are 3 columns and 12 rows. In the first row insert n , xn , yn , move to the second row and insert 0, 0, 1. That last value is $y(0)$.

First column. Highlight the first column from **A2** to **A12**, and then click on the button on the extreme left of the third row of the menus at the top of the page. The button looks like a window with three vertical panes and the blind pulled down over the top pane, along with an arrow pointing down. In the menu box that appears, insert a step size of 1 and a stop value of 10.

Second column. Highlight the second column from **B2** to **B12**, click on the button above on the extreme left (same as before), and insert a step size of .1 and a stop value of 1.

Third column. Carefully insert into **C3**: $\sim C2 + (.1) * f(\sim B2, \sim C2)$ and <Enter>. Note how this entry will reflect the old value of y plus the slope times the change in x , as it should. To complete this column, highlight from **C3** to **C12**, click on **Spreadsheet**, **Fill**, and **Down**. The spreadsheet should look like the one displayed below.

	A	B	C
1	n	xn	yn
2	0	0	1
3	1	.1000	1.1000
4	2	.2000	1.2200
5	3	.3000	1.3620
6	4	.4000	1.5282
7	5	.5000	1.7210
8	6	.6000	1.9431
9	7	.7000	2.1974
10	8	.8000	2.4871
11	9	.9000	2.8158
12	10	1	3.1874

There should be a prompt > below the spreadsheet. Enter **M:=** and then highlight the region in the spreadsheet from **B2** to **C12**, click on the **Copy** button at the top, move the cursor to after the **M:=** on the line below and then click on the **Paste** button at the top. Insert a colon at the end of that line, and then <Enter>. You should see:

```
> M:=MATRIX([[0, 1], [.1, 1.1], [.2, 1.22], [.3, 1.362], [.4,
1.5282], [.5, 1.72102], [.6, 1.943122], [.7, 2.1974342], [.8,
2.48717762], [.9, 2.815895382], [1, 3.187484920]]):
>
```

Now we must pair the points together by converting this *matrix* of 1×2 matrices to a *list* of 1×2 matrices. Then we will be able to treat each 1×2 matrix as a point in the plane and plot the graph.

```
> a:=convert(M,listlist);
a:=[[0,1], [.1,1.1], [.2,1.22], [.3,1.362], [.4,1.5282], [.5,1.72102], [.6,1.943122],
[.7,2.1974342], [.8,2.48717762], [.9,2.815895382], [1,3.187484920]]
```

We are going to plot these points and join them together with line segments, and then display the plot at the same time that we display the direction field. This means that we must give these plots names and suppress their output until we are ready.

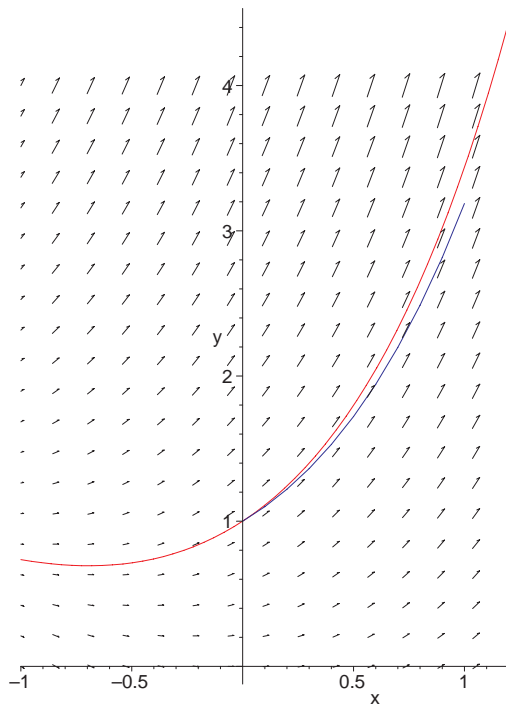
```
> A:=pointplot(a,style=line,color=blue):
```

Now to construct a function that Maple can interpret as acceptable for the command **fieldplot**:

```
> F:=(x,y)->[1,f(x,y)];
                                      $F := (x, y) \rightarrow [1, f(x, y)]$ 
> B:=fieldplot(F(x,y),x=-1..2,y=0..4):
```

Because we happen to know the solution to this equation we will include it in the plot.

```
> C:=plot(2*exp(x)-x-1,x=-1..2,color=red):
> display(A,B,C);
```



C2M8 Problem Use Maple to graph an approximation to the solution of the differential equation

$$y' = x - y^2 \quad y(1) = -1$$

Use a step size of 0.1 and approximate $y(3)$. This requires twenty steps. The work you submit should reflect the Maple Example above and should display the spreadsheet and display the approximation on the same plot with the direction field. Do not be concerned about the precise solution.